



PATENT ABSTRACTS OF JAPAN

(11)Publication number : 06-332799

(43)Date of publication of application : 02.12.1994

(51)Int.Cl.

G06F 12/06

G06F 15/40

(21)Application number : 05-145412

(71)Applicant : NEC CORP

(22)Date of filing : 26.05.1993

(72)Inventor : WATABE HAJIME

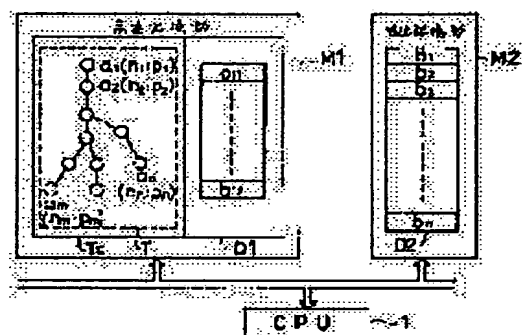
(54) TREE INFORMATION STORAGE SYSTEM

(57)Abstract:

PURPOSE: To shorten data processing time by reducing the number of times of access to a low speed storage part when the processing of data having a tree structure is performed.

CONSTITUTION: When each data having a tree structure is accessed in accordance with a tree structure, the number of times of access is calculated every data, the reference probability of each data is determined based on this number of times of access the data with high reference probability of the data stored in a low speed storage part M2 is selected and the data is stored in a reading high speed storage part M1.

Therefore, when a data processing is performed by referring to each data, the access to the low speed storage part M2 can be reduced and the data processing can be performed at high speed.



LEGAL STATUS

[Date of request for examination] 26.05.1993

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number] 2560610

[Date of registration] 19.09.1996

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right] 19.09.2001

Copyright (C); 1998,2003 Japan Patent Office

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平6-332799

(43) 公開日 平成6年(1994)12月2日

(51) Int.Cl.⁵

G 0 6 F 12/06
15/40

識別記号

5 2 0
5 0 0 C

庁内整理番号

9368-5B
9194-5L

F I

技術表示箇所

審査請求 有 請求項の数 1 F D (全 6 頁)

(21) 出願番号 特願平5-145412

(22) 出願日 平成5年(1993)5月26日

(71) 出願人 000004237

日本電気株式会社

東京都港区芝五丁目7番1号

(72) 発明者 渡部 元

東京都港区芝五丁目7番1号 日本電気株式会社内

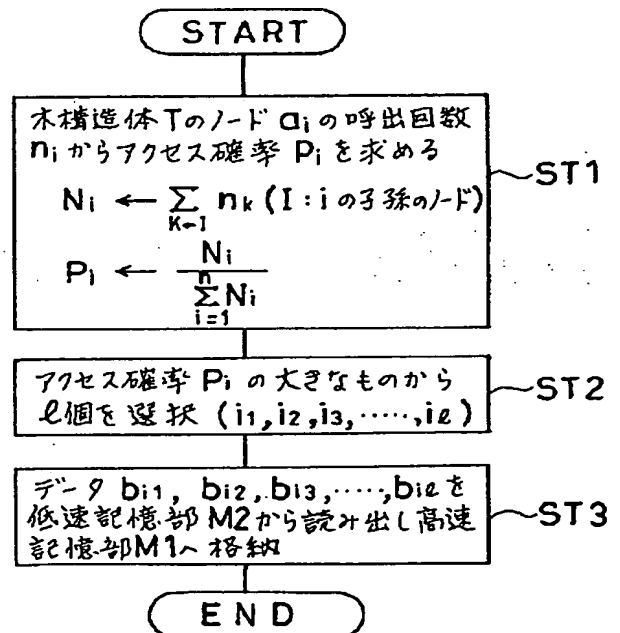
(74) 代理人 弁理士 山川 政樹

(54) 【発明の名称】 木情報格納方式

(57) 【要約】

【目的】 木構造を有するデータの処理を行う場合に、低速記憶部へのアクセス回数を減らしてデータ処理時間を短縮する。

【構成】 木構造を有する各データが木構造に従ってアクセスされる場合に、各データ毎にアクセス回数を算出すると共に、このアクセス回数に基づいて各データの参照確率を求め、低速記憶部M2に格納されているデータのうち参照確率の高いデータを選択して読み出し高速記憶部M1に格納する。したがって、各データを参照してデータ処理を行う場合に低速記憶部へのアクセスを少なくでき、高速でデータ処理を行うことができる。



【特許請求の範囲】

【請求項1】 高速記憶部及び低速記憶部を備え、前記各記憶部の何れか一方に格納された木構造を有する各データを前記木構造にしたがってアクセスしデータ処理を行うデータ処理装置において、

各データを前記木構造にしたがってアクセスした場合のアクセス回数を各データ毎に算出するアクセス回数算出手段と、算出された各データのアクセス回数に基づいて各データの参照確率を算出する参照確率算出手段と、前記低速記憶部に格納されたデータのうち参照確率の高いデータを1つ選択して前記高速記憶部へ格納する手段とを備えたことを特徴とする木情報格納方式。

【発明の詳細な説明】

【0001】

【産業上の利用分野】 本発明は、木構造にしたがってアクセスされるデータを格納する木情報格納方式に関する。

【0002】

【従来の技術】 従来のこの種の木情報格納方式を適用した装置は、図3に示すように、CPU1、高速記憶部M1、低速記憶部M2から構成されている。ここで、低速記憶部M2のデータ格納エリアD2には各データb1、b2、b3、・・・、bnが格納されている。また、これらの各データには各々ラベルが付与されこれらのラベルは上記各データに対応する各ノードa1、a2、a3、・・・、anとして高速記憶部M1に格納される。

なお、これらの各ノードa1、a2、a3、・・・、anは、木構造となっており、これらのノードは木構造体Tとして高速記憶部M1に格納される。ここで本装置のデータ参照方式は、各ノードが木構造となっていることから、或ノードに対応するデータが参照されると、これより先祖のノード、つまり「葉」から「根」のノードに相応するデータへと順次参照されるものである。即ち、ノードakに対応するデータbkが参照されると、先祖のノードであるノードa1の方向へ順次参照が行われる。そしてこのとき対応するデータbkは低速記憶部M2のエリアD2から読みだされ高速記憶部M1のデータ格納エリアD1に格納される。

【0003】 図4は、このような従来装置のCPU1の動作を示すフローチャートである。即ち、CPU1がステップST11で木構造体Tのノードakを選択したとする。するとステップST12では、高速記憶部M1のデータ格納エリアD1の中からノードakに対応するデータbkの検索を開始する。次いで、ステップST13では高速記憶部M1内のエリアD1中にデータbkの有無を判断し、データbkがエリアD1に存在しなければステップST14で低速記憶部M2のデータ格納エリアD2から該当のデータを読みだし高速記憶部M1のデータ格納エリアD1へ格納する。続いてステップST15ではデータ格納エリアD1上のデータbkを参照する。

【0004】 次にステップST16では、「k=1」つまりデータの参照がノードa1まで行われたか否かを判断し、これが「N」となると、ノードa1の方向にノードが選択されるように次のノードをステップST17で選択する。そしてステップST12へ戻り、選択されたノードに対応するデータの参照を行う。このようなデータの参照動作は、ノードa1に対応するデータb1の参照が終了するまで繰り返し反復実行される。ここで、低速記憶部M2のデータbkを読み出して高速記憶部M1に格納する際には、格納エリアD1上の旧データを1つつぶしてこのエリアに格納しなければならない。この場合、つぶされる旧データは、このデータにアクセスした時刻に基づいて決定される。

【0005】

【発明が解決しようとする課題】 従来は、各ノードに対応するデータを参照しようとする際には、データが高速記憶部M1に存在しなければ低速記憶部M2から読み出すようにしており、低速記憶部M2へのアクセスは時間がかかることから、データの処理時間が長くなるという欠点があった。

【0006】 したがって本発明は、木構造を有するデータの処理を行う場合に、低速記憶部へのアクセス回数を減らしてデータ処理時間を短縮することを目的とする。

【0007】

【課題を解決するための手段】 このような課題を解決するために本発明は、各データを木構造に従ってアクセスした場合のアクセス回数を各データ毎に算出するアクセス回数算出手段と、算出された各データのアクセス回数に基づいて各データの参照確率を算出する参照確率算出手段と、低速記憶部に格納されたデータのうち参照確率の高いデータを選択して高速記憶部へ格納する手段とを設けたものである。

【0008】

【作用】 木構造を有する各データが木構造にしたがってアクセスされる場合に、各データ毎にアクセス回数が算出されると共に、アクセス回数に基づいて各データの参照確率が求められ、参照確率の高いデータが高速記憶部に格納される。したがって、各データを参照してデータ処理を行う場合に低速記憶部へのアクセスを少なくでき、高速でデータ処理を行うことができる。

【0009】

【実施例】 以下、本発明について図面を参照して説明する。図1は本発明に係る木情報格納方式を適用した装置のブロック図である。ここで、図1に示す装置の構成は上記した図3に示す従来構成とほぼ同一の構成となっているが、従来例とは、高速記憶部M1内のデータ格納エリアD1に対するデータの格納方法の点が異なっている。

【0010】 ところで、低速記憶部M2のデータ格納エリアD2には本装置の全てのデータb1、b2、b3、

3

・・・, b_n が格納されていると共に、これらの各データに対応した各ノード $a_1, a_2, a_3, \dots, a_n$ は、例えば「根」に相当するノード a_1 から「葉」に相当するノード a_n までの木構造体 T として高速記憶部 M_1 内の木構造体格納エリア T_E に格納されている。したがって、各データを検索する場合、本装置は、各データに対応する各ノードが木構造となっていることから、この木構造にしたがい順次データの検索を行うように構成されている。従来は、高速アクセス可能な高速記憶部 M_1 内のデータ格納エリア D_1 に格納されるデータは、木構造とは無関係に格納されており、このため木構造にしたがってデータを検索した場合、上記エリア D_1 には該当するデータが存在せずに高速アクセス不可能な低速記憶部 M_2 内のデータ格納エリア D_2 がしばしばアクセスされ、データ処理に時間がかかる欠点を有していた。これに対して、本装置では、高速アクセスが可能なデータ格納エリア D_1 にアクセス頻度の高いデータを格納するものである。

【0011】一般に、上記した「根」ノードと「葉」ノードとの関係、即ち「親」ノードと「子」ノードとの関

$$N_i = \sum_{k \leftarrow I} n_k$$

【0014】ここで式①中の I は、ノード a_i を「親」ノードとした場合、「子孫」に相当するノードを意味している。即ち、各ノードは木構造を有していることからノード a_i の総アクセス回数 N_i は、ノード a_i のみのアクセス回数 n_i に、子孫のノードがアクセスされることによりアクセスされる回数を加えた総和として表される。

【0015】こうしてノード a_i の総アクセス回数 N_i ※30

$$P_i = N_i / \sum_{i=1}^x N_i \quad \dots \dots \dots \textcircled{2}$$

【0017】こうして、ノード a_i のアクセス確率 P_i が算出されると、同様に各ノードのアクセス確率を算出する。次にステップ ST_2 では、算出された各ノードのアクセス確率のうち、値の大きいものから順に 1 (エル) 個 ($i_1, i_2, i_3, \dots, i_l$) 選択する。そしてステップ ST_3 では、選択されたアクセス確率を有するノードに対応するデータ ($b_{i1}, b_{i2}, b_{i3}, \dots, b_{il}$) を、低速記憶部 M_2 のデータ格納エリア D_2 から読みだして高速記憶部 M_1 のデータ格納エリア D_1 へ格納する。

【0018】このようにしてアクセス頻度の高いデータを高速記憶部 M_1 のエリア D_1 へ予め格納しておくことにより、データ参照の際には高速アクセス可能なエリア D_1 がアクセスされることになり、データの処理時間を従来例と比べて大幅に短縮することができる。なお、データ参照の際にエリア D_1 にデータが存在しない場合は、低速記憶部 M_2 のデータ格納エリア D_2 を検索して

4

*係を有する木構造データは、「子」ノードに対応するデータが参照されると、必ず「親」ノードに対応するデータも参照される関係にあり、したがって、「親」ノードのデータ参照確率は「子」ノードのデータ参照確率よりも高くなる。本装置では、このような各ノードが参照される確率を予め計算し、この確率の高いものから順番に上記高速記憶部 M_1 のエリア D_1 に格納することにより、高速アクセスが不可能な低速記憶部 M_2 内のデータ格納エリア D_2 へのアクセス回数を減らし、データ処理時間を短縮するようにしたものである。

【0012】次に、上記実施例装置の CPU_1 の動作を図2のフローチャートに基づいて説明する。本実施例装置では、参照される高速記憶部 M_1 のデータ格納エリア D_1 のデータは、予め次の手順により設定される。即ち、ステップ ST_1 として、ノードを a_i とした場合、まずノード a_i の呼出回数 (即ち、アクセス回数) N_i を求める。即ち、

【0013】

【数1】

※が算出されると、同様に各ノードについてそれぞれその総アクセス回数を求め、さらに全ノードのアクセス回数の総和 (後述の式②中の分母に相当) を算出する。次に、ノード a_i の総アクセス回数 N_i と全ノードのアクセス回数の総和とからノード a_i のアクセス確率 P_i を次式にしたがって算出する。即ち、

【0016】

【数2】

このエリアから該当のデータを読み出すが、この際、読み出したデータは高速記憶部 M_1 のエリア D_1 には格納しない。

【0019】

【発明の効果】以上説明したように本発明によれば、木構造を有する各データが木構造にしたがってアクセスされる場合に、各データ毎にアクセス回数を算出すると共に、このアクセス回数に基づいて各データの参照確率を求め、参照確率の高いデータを高速記憶部に格納するようにしたので、各データを参照してデータ処理を行う場合に低速記憶部へのアクセスを少なくでき、したがって高速でデータ処理が行えるという効果がある。

【図面の簡単な説明】

【図1】本発明に係る木情報格納方式を適用した装置の一実施例を示すブロック図である。

【図2】上記実施例装置の動作を示すフローチャートである。

5

6

【図3】従来装置のブロック図である。

【図4】従来装置の動作を示すフローチャートである。

【符号の説明】

1 CPU

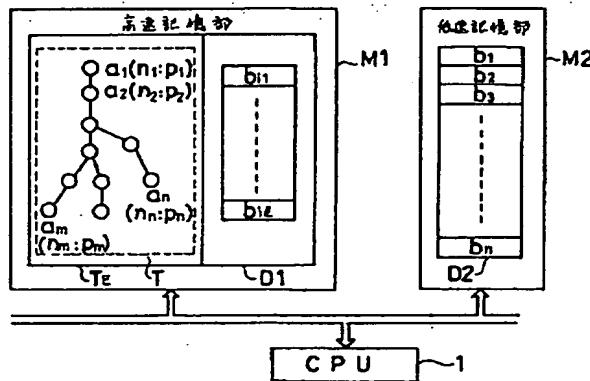
M1 高速記憶部

M2 低速記憶部

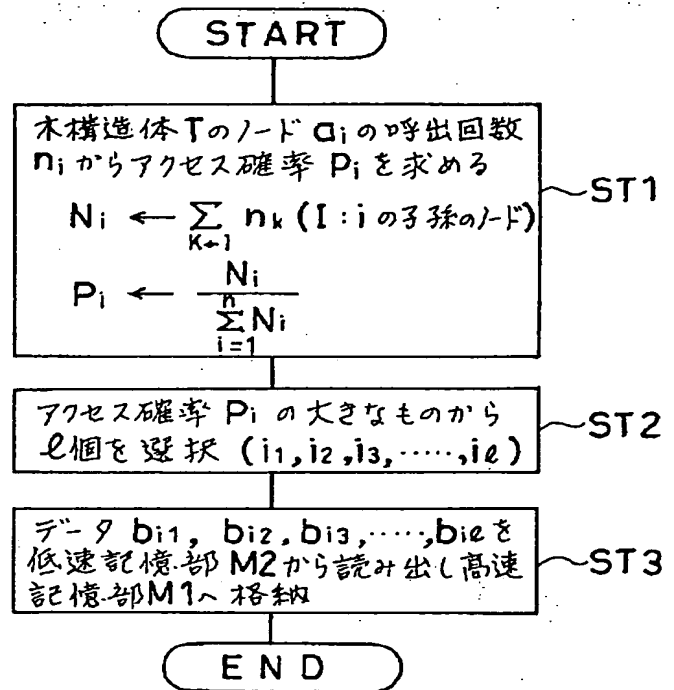
D1, D2 データ格納エリア

TE 木構造体格納エリア

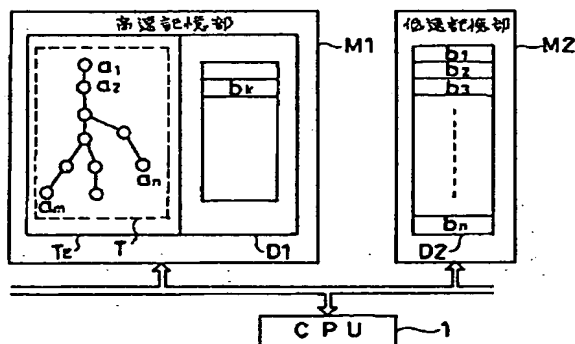
【図1】



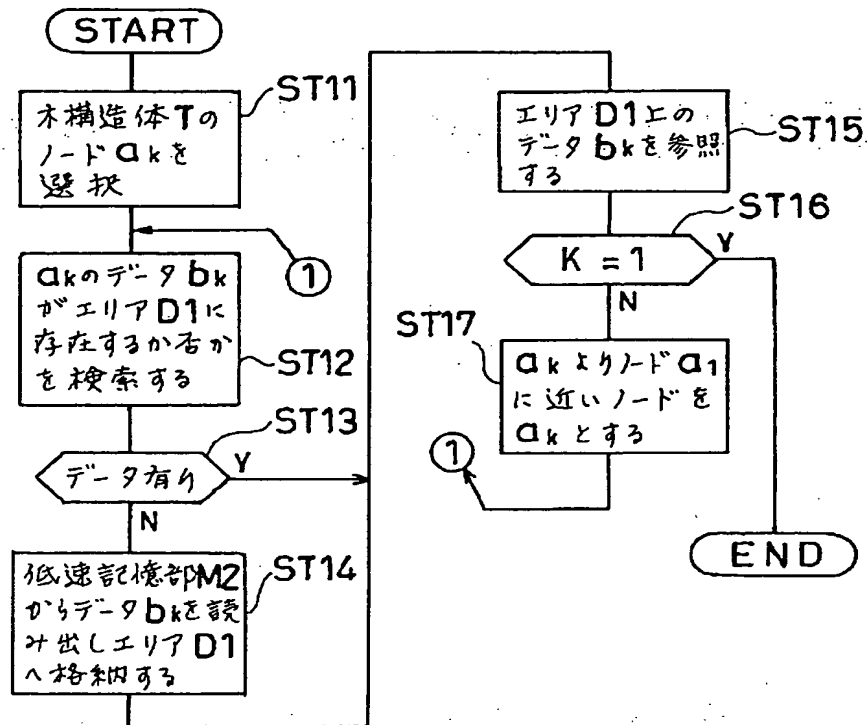
【図2】



【図3】



【図4】



【手続補正書】

【提出日】平成6年5月12日

【手続補正1】

【補正対象書類名】明細書

【補正対象項目名】特許請求の範囲

【補正方法】変更

【補正内容】

【特許請求の範囲】

【請求項1】 高速記憶部及び低速記憶部を備え、前記各記憶部の何れか一方に格納された木構造を有する各データを前記木構造にしたがってアクセスしデータ処理を行うデータ処理装置において、各データを前記木構造にしたがってアクセスした場合のアクセス回数を各データ毎に算出するアクセス回数算出手段と、算出された各データのアクセス回数に基づいて各データの参照確率を算出する参照確率算出手段と、前記低速記憶部に格納されたデータのうち参照確率の高いデータを選択して前記高速記憶部へ格納する手段とを備えたことを特徴とする木情報格納方式。

【請求項2】 請求項1において、

前記アクセス回数算出手段によるアクセス回数の算出は、あるノードのみのアクセス回数にそのノードの葉に対するノードのアクセス回数を加算した総和としてその

ノードの総アクセス回数を求め、

前記参照確率算出手段によるアクセス確率の算出は、前記ノードの総アクセス回数を全ノードについて算出して全ノードアクセス回数総和を求め、前記あるノードの総アクセス回数をこの全ノードアクセス回数総和で除してそのノードのアクセス確率として求めることを特徴とする木情報格納方式。

【手続補正2】

【補正対象書類名】明細書

【補正対象項目名】0007

【補正方法】変更

【補正内容】

【0007】

【課題を解決するための手段】このような課題を解決するために本発明は、各データを木構造に従ってアクセスした場合のアクセス回数を各データ毎に算出するアクセス回数算出手段と、算出された各データのアクセス回数に基づいて各データの参照確率を算出する参照確率算出手段と、低速記憶部に格納されたデータのうち参照確率の高いデータを選択して高速記憶部へ格納する手段とを設けたものである。また、アクセス回数算出手段によるアクセス回数の算出は、あるノードのみのアクセス回数

にそのノードの葉に対するノードのアクセス回数を加算した総和としてそのノードの総アクセス回数を求め、参照確率算出手段によるアクセス確率の算出は、前記ノードの総アクセス回数を全ノードについて算出して全ノード

ドアクセス回数総和を求め、前記あるノードの総アクセス回数をこの全ノードアクセス回数総和で除してそのノードのアクセス確率として求めるものである。